

REMARKS

Claims 1-24 are pending in the application. Claims 1-24 have been rejected. Claims 2, 10, and 18 have been cancelled.

Claims 1-24 stand rejected under 35 U.S.C. § 112, second paragraph, based upon the use of the term “cluster.” This rejection is respectfully traversed. The term “cluster” is clearly defined within the specification, which sets forth, for example:

By way of introducing terminology, a cluster is a set of two or more interacting entities. A cluster of virtual machines may be referred to as a virtual machine cluster, a virtual machine grid, or a logical virtual machine. A single virtual machine that contains functionality for operating in a cluster/grid may be referred to as a clusterable virtual machine or a grid-enabled virtual machine. A single virtual machine that is included within a cluster/grid may be referred to as a clustered virtual machine or a grid-embedded virtual machine (Broussard Application ¶[0045]).

Accordingly, a “cluster” is a set of two or more interacting entities.

Claims 9-24 stand rejected under 35 U.S.C. § 101 as directed toward non-statutory subject matter. The specification has been amended to address this rejection.

Claims 1, 3-7, 9, 11-15, 17 and 19-23 stand rejected under 35 U.S.C. § 102(e) as being anticipated by van Rietschote et al., U.S. Patent No. 7,213,246 (Rietschote). Claims 2, 10 and 18 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Rietschote in view of Dillenberger et al., U.S. Patent No. 6,732,139 (Dillenberger). Claims 8, 16 and 24 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Rietschote in view of Aridor et al., U.S. Patent No. 6,618,737 (Aridor). These rejections are respectfully traversed.

Claims 1, 2-7, 9, 11-15 and 19-23 are allowable over Rietschote and Dillenberger.

The present invention, as set forth by independent claim 1 relates to a method for operating a virtual machine within a data processing system. The method includes the computer-implemented steps of running a plurality of virtual machines on one or more devices within the data processing system, wherein each virtual machine in the plurality of virtual machines incorporates functionality for interoperating with other virtual machines in a virtual machine

cluster, associating the plurality of virtual machines in a virtual machine cluster, wherein each virtual machine in the virtual machine cluster acts as a node within the virtual machine cluster, and sharing information about the plurality of virtual machines within the virtual machine cluster such that a virtual machine may be added to the virtual machine cluster or such that a virtual machine may be removed from the virtual machine cluster as the plurality of virtual machines continues to run.

The present invention, as set forth by independent claim 9 relates a computer program product on a computer readable medium for use within a data processing system for operating a virtual machine. The computer program product includes means for running one of a plurality of virtual machines on one or more devices within the data processing system, wherein each virtual machine in the plurality of virtual machines incorporates functionality for interoperating with other virtual machines in a virtual machine cluster, means for associating a virtual machine with the plurality of virtual machines in a virtual machine cluster, wherein each virtual machine in the virtual machine cluster acts as a node within the virtual machine cluster, and means for sharing information about the plurality of virtual machines within the virtual machine cluster such that a virtual machine may be added to the virtual machine cluster or such that a virtual machine may be removed from the virtual machine cluster as the plurality of virtual machines continues to run.

The present invention, as set forth by independent claim 17 relates to an apparatus within a data processing system for operating a virtual machine. The apparatus includes means for running one of a plurality of virtual machines on one or more devices within the data processing system, wherein each virtual machine in the plurality of virtual machines incorporates functionality for interoperating with other virtual machines in a virtual machine cluster, means for associating a virtual machine with the plurality of virtual machines in a virtual machine cluster, wherein each virtual machine in the virtual machine cluster acts as a node within the virtual machine cluster, and means for sharing information about the plurality of virtual machines within the virtual machine cluster such that a virtual machine may be added to the virtual machine cluster or such that a virtual machine may be removed from the virtual machine cluster as the plurality of virtual machines continues to run.

Rietschote generally disclose a cluster of computer systems where the computer systems may include virtual machines. (See e.g., Rietschote, Col. 2, lines 46 – 65.)

When discussing Rietschote, the Examiner set forth:

Rietschote does not explicitly disclose of sharing information about the plurality of virtual machines within the virtual machine cluster such that a virtual machine can be added to the virtual machine cluster or such that a virtual machine may be removed from the virtual machine cluster as the plurality of virtual machines continues to run. However, Dillenberger discloses a method in which a virtual machine can be added or removed from a virtual machine cluster while the plurality of virtual machines continue to run (Office Action dated December 11, 2007, ¶19, Page 8).

Dillenberger discloses a distributed Java virtual machine method for managing distributed heterogeneous network nodes to meet demand for service requests. Upon receipt of a request for service, a Java workload manager (JWLM) classifies the request into a particular type of service and places the work on a logical queue to be executed. Routing queues place the work on one of many available nodes. The work is distributed to the nodes based on accounting, availability, and past performance of all nodes considered.

The portion of Dillenberger to which the Examiner refers sets forth:

The node manager manages a node and dynamically adds and removes new virtual machines on the node for executing the work of portion of the work (Dillenberger, Col. 8, lines 1-3).

However, neither Rietschote or Dillenberger disclose or suggest *sharing information about the plurality of virtual machines within a virtual machine cluster* such that a virtual machine may be added to the virtual machine cluster or such that a virtual machine may be removed from the virtual machine cluster as the plurality of virtual machines continues to run, as is required by claim 1 and as is substantially required by claims 9 and 17. Accordingly, claims 1, 9 and 17 are allowable over Rietschote and Dillenberger.

More specifically, Rietschote and Dillenberger, taken alone or in combination, do not teach or suggest a method for operating a virtual machine within a data processing system which includes the computer-implemented steps of running a plurality of virtual machines on one or more devices within the data processing system, wherein each virtual machine in the plurality of virtual machines incorporates functionality for interoperating with other virtual machines in a

virtual machine cluster, and associating the plurality of virtual machines in a virtual machine cluster, wherein each virtual machine in the virtual machine cluster acts as a node within the virtual machine cluster and sharing information about the plurality of virtual machines within the virtual machine cluster such that a virtual machine may be added to the virtual machine cluster or such that a virtual machine may be removed from the virtual machine cluster as the plurality of virtual machines continues to run, all as required by claim 1. Accordingly, claim 1 is allowable over Rietschote and Dillenberger. Claims 3-8 depend from claim 1 and are allowable for at least this reason.

Rietschote and Dillenberger, taken alone or in combination, do not teach or suggest a computer program product on a computer readable medium for use within a data processing system for operating a virtual machine which includes means for running one of a plurality of virtual machines on one or more devices within the data processing system, wherein each virtual machine in the plurality of virtual machines incorporates functionality for interoperating with other virtual machines in a virtual machine cluster, means for associating a virtual machine with the plurality of virtual machines in a virtual machine cluster, wherein each virtual machine in the virtual machine cluster acts as a node within the virtual machine cluster and means for sharing information about the plurality of virtual machines within the virtual machine cluster such that a virtual machine may be added to the virtual machine cluster or such that a virtual machine may be removed from the virtual machine cluster as the plurality of virtual machines continues to run, all as required by claim 9. Accordingly, claim 9 is allowable over Rietschote and Dillenberger. Claims 11-16 depend from claim 9 and are allowable for at least this reason.

Rietschote and Dillenberger, taken alone or in combination, do not teach or suggest an apparatus within a data processing system for operating a virtual machine which includes means for running one of a plurality of virtual machines on one or more devices within the data processing system, wherein each virtual machine in the plurality of virtual machines incorporates functionality for interoperating with other virtual machines in a virtual machine cluster, means for associating a virtual machine with the plurality of virtual machines in a virtual machine cluster, wherein each virtual machine in the virtual machine cluster acts as a node within the virtual machine cluster and means for sharing information about the plurality of virtual machines within the virtual machine cluster such that a virtual machine may be added to the virtual

machine cluster or such that a virtual machine may be removed from the virtual machine cluster as the plurality of virtual machines continues to run., all as required by claim 17. Accordingly, claim 17 is allowable over Rietschote and Dillenberger. Claims 19-24 depend from claim 17 and are allowable for at least this reason.

Additionally, Rietschote and Dillenberger do not disclose or suggest *sharing load values representing computer resource utilization* among the virtual machines *in the virtual machine cluster* and performing a load-balancing operation across the virtual machine cluster, as is required by claim 3 and as is substantially required by claims 11 and 19. Accordingly, claims 3, 11 and 19 are allowable over Rietschote and Dillenberger.

Claims 8, 16 and 24 are allowable over Rietschote and Aridor.

Rietschote is discussed above.

Aridor discloses a technique of field-level caching in distributed object-oriented systems, in which a speculative approach is taken to identify opportunities for caching. Invalidation protocols, which are fully compliant with a Java memory model, are provided to recover from incorrect speculation. The technique is implemented on a cluster of machines, and is scalable to multi-threaded applications. The technique distributes an application's threads among nodes of a cluster. While Aridor discloses distributing among nodes of a cluster, Aridor is silent on dispatching threads to different virtual machines such that the execution of the multi-threaded application spans virtual machines.

Accordingly, in addition to the distinctions discussed above, Rietschote or Aridor do not disclose or suggest *dispatching threads of the multi-threaded application on different virtual machines such that execution of the multi-threaded application spans multiple virtual machines*, as is required by claim 8 and as is substantially required by claims 16 and 24. Accordingly, claims 8, 16 and 24 are allowable over Rietschote and Aridor..

CONCLUSION

In view of the amendments and remarks set forth herein, the application is believed to be in condition for allowance and a notice to that effect is solicited. Nonetheless, should any issues remain that might be subject to resolution through a telephonic interview, the examiner is requested to telephone the undersigned.

I hereby certify that this correspondence is being electronically submitted to the COMMISSIONER FOR PATENTS via EFS on March 11, 2008.

/Stephen A. Terrile/

Attorney for Applicant(s)

Respectfully submitted,

/Stephen A. Terrile/

Stephen A. Terrile
Attorney for Applicant(s)
Reg. No. 32,946